In the Specification

Please amend the specification as follows:

Page 1, lines 3-5 have been rewritten as follows:

--The present invention relates to a MPEG-2 transport decoder, and more particularly, to a device and method for filtering added information, in which PSI (Program Specification Information) in a transport stream is processed in an MPEG-2 system layer.--

Page 1, line 7 through page 2, line 19, have been rewritten as follows:

--The MPEG-2, widely known as a moving picture compression method, has standards on a system layer for compression of an audio data, in addition to the moving picture data, and changing the moving picture data and the audio data suitable for transmission. The system layer has two systems, one is a TS (Transport Stream), and the other is a PS (Program Stream). While the PS system forms a pack by grouping a plurality of PES (Packetized Elementary Stream), reversely the TS system divides the PES for loading on a plurality of transport packets before transmission. That is, since the TS system transmits a plurality of video and audio individual bitstreams, the TS system requires information on selection of a program from a plurality of programs, and on selection of a packet, and on how to decode the packet, that is called as PSI (Program Specification Information). The PSI is transmitted by a packet having

identification codes or a packet indicated by a primary PSI. In the MPEG-2 system layer, the PSI is one of elements in the TS. In decoding and reproduction of the TS, one of the plurality of programs should be selected, and PIDs (Packet IDentification numbers) of the transport packets of the individual bitstreams required for the decoding and reproduction of the program should be known. Then, parameter information or linkage information of the individual bitstreams should be known. In order to conduct the foregoing multistage operation, a plurality of added information tables are required, which are transmitted by a data structure called as sections. Of the plurality of tables, PAT (Program Association Table) is special information transmitted by a packet with PID=0. Every program number in the PAT has a description on elements of a relevant program, and particularly, has a PID of the transport packet which transmits a PMT (Program Map Table). The PMT describes a program PID, and a PID list and annex information of a transport packet which transmits individual video and audio bitstreams. In this instance, different from the PES which deals with ES (Elementary Stream), coded video or audio data, the PSI is provided with a redundancy for making fast access available since the PSI includes information on the program. That is, the PSI has identical information repeatedly for quick starting of decoding at any time. For example, standards of the ATSC (Advanced Television Systems Committee) requires transmission of information on PAT within at least 0.1 seconds, and on PMT within 0.4 seconds. That is, if the PAT is not available, there can be a

problem in decoding since the information can not be identified as being a video or an audio. However, because there is no change in the information in most of the cases, information identical to the one transmitted before is transmitted. Accordingly, a general transport decoder has an algorithm for effective processing of such repeated data, one of which known widely is the section filtering defined in a DVB (Digital Video Broadcasting) standard. That is, the MPEG-2 has standards for four PSI tables each having a basic unit called a section, a combination of which forms one table. Information may be transmitted, with the information put in one section or a plurality of sections.--

On page 5, lines 8-20 have been rewritten as follows:

FIG. 1 illustrates one example of an 8 byte section filter, wherein, in a case mask bit is '1', a section data of a corresponding byte (or bit) is compared to a match data, for processing only sections which match the match data. 'A' type section filter in FIG. 1 illustrates a case when the mask is set up in bit units, and 'B' type section filter in FIG. 1 illustrates a case when the mask is set up in byte units. For example, it can be known that a comparison of data in the section 'A' in (a) in FIG. 1 to the match data in the 'A' type section filter with a mask bit of '1' in (c) indicates that the two data are the same. In this case it is determined that the two data are matched, and the data in the section 'A' is received and stored in a designated position. In this instance, since the data in the section 'A' and the match data in the 'B' type sector filter

with a mask byte of '1' in (d) are not the same, the B type section filter does not receive the data in the section A. That is, it can be known that the section A in (a) and the A type section filter in (c) are matched, and the section B in (b) and the B type section filter in (d) are matched. Of PSI sections processed and transmitted thus, a desired PSI can be selected and processed. In fact, a DVB standard requires at least 32 section filters each with at least 8 bytes.--

On page 7, line 23 through page 8, line 2 have been rewritten as follows:

--The method further includes the step of skipping the section received at the present time if it is determined in the step (1) that there are no table IDs matched, or if it is determined in the step (2) that the two version numbers are the same.--

On page 8, lines 5-9 have been rewritten as follows:

--The method further includes the steps of (6-1) determining completion of the table of the version processed at the present time, if it is determined in the step (1) that the table ID of the section received at the present time and the any one of table IDs stored in the memory are matched, and (6-1) mask enabling the version number of the section if it is determined in the (6-1) step that the table is completed.--

On page 9, line 7 through page 11, line 12, have been rewritten as follows:

--Reference will now be made in detail to the preferred embodiments of the present invention, examples of which are illustrated in the accompanying drawings. Device and method for filtering added information of the present invention suggests application of only table IDs and version numbers to a filter for reducing a size and complexity of hardware, particularly memories. The table IDs are assigned to every table, and are included in every section. The device and method for filtering added information of the present invention also suggests receiving a data which has the same table ID, but a different version number. FIG. 2 illustrates a system of hardware basis embodiment of a method for filtering added information in accordance with a preferred embodiment of the present invention, and FIG. 3 illustrates a flow chart showing a method for filtering added information in accordance with a first preferred embodiment of the present invention, schematically.

Referring to FIG 2, the system includes a table ID memory 22 for storing table IDs of sections, and a memory 21 for masking enable bits E for each table ID and versions to be masked. If a section 20 is received, a start of the section is searched for (S 301). If a payload unit_start_indicator is '1' in a TS packet having a PSI payload, the start of the section can be known from a pointer field. And, another section is started if the next data is not 0xFF after one section is ended. In this instance, a first byte of the section 20 is the table ID.

Therefore, the table ID is compared to the table IDs stored in the table ID memory 22 (S302). If there is a table ID match, a version number assigned to the matched table ID is read from the version memory 21 for the table ID; if not, the present section is skipped. In the meantime, if there is a matched table ID, it is determined whether or not a masking enable signal 'E' is '1'. If the masking enable signal 'E' is '1', a version number in the section is compared to a version number in the version memory 21 (S 303). If the version of the present section is the same with the version in the version memory 21, and in a state of mask enable, the present version is masked. That is, the same data has been received and stored, already. Therefore, in this instance, the present data is skipped (S 304). The masking enable signal 'E' is a flag for determining storage of the received section. If the present section in the step S 303 is different from the version in the version memory 21 and there is no mask enable signal, the present section is either stored or processed (S 305).

FIG. 4 illustrates a flow chart showing a method for filtering added information in accordance with a second preferred embodiment of the present invention, wherein a start of a received section is searched for in the same method with the one shown in FIG. 3 (S 401). Once the start of the section is found in the step 401, matching of the table ID in the located section and the table ID in the memory 22 is determined (S 402). If it is found in the step S 402 that the two table IDs are matching, the version number stored in the version memory 21 for the table ID and the version number of the located

section are compared to each other (S 403). If it is found in the step S 403 that the version of the present section is the same with the version in the version memory 21, and the mask enable bit 'E' is '1', the present section is skipped (S 404) as the present version is masked (S 404). If it is found in the step S 403 that the version of the present section is different from the version in the version memory 21, and it is not in a mask enable state, the present section is stored or processed (S405). In the meantime, if it is determined in the step S402 that the table IDs are matching, completion of one table is determined (S406). The table is deemed completed if all sections of the table for the version processed presently are processed or no more sections are required. That is, completion of the table is verified on reception of section_number and last_section_number from the received section. If it is determined in the step S406 that the table is completed, a table completion bit 'C' of the table is set to '1', and if a version number automatic setting bit 'A' is '1', a received version number is stored in the version memory 21. The table completion bit 'C' is a flag set automatically for not processing the same sections of the version received after all the sections of one version is processed. And, the version number automatic setting bit 'A' is a flag for automatic updating of the version stored in the version memory to a new version when sections of a new version is received. That is, if it is intended to store a received version number in the memory 21, the version number is stored after the mask enable bit 'E' is set to '1' if the table completion bit 'C' is '1'.--

On page 12, lines 11-21 have been rewritten as follows:

--In a case of the related art section filtering, the host sets the filter such that the sections with version numbers 0×00 are processed at first, and then the section with version number 0×01 is processed. And, upon reception of the section with the version number 0×01, the section is stored through the filter. The host sets the filter such that the stored section is read and decoded, and, then, the sections with version numbers 0×02 are processed. However, in the aforementioned case, the long time period required for setting the filter for processing the sections with version numbers 0×02 may lead to miss a first section of the version number 0×02. However, since the present invention suggests to mask the section with version number 0×01 at the hardware as soon as the section with version number 0×01 is received, the next sections with version numbers 0×02 can be processed without missing any of the sections with version numbers 0×02.--